Reconocimiento de Patrones

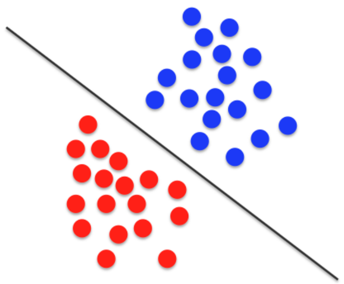
Version 2022-2

SVM

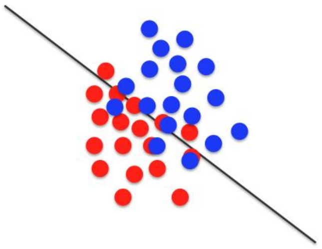
Dr. José Ramón Iglesias
DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingenieria Electronica
Universidad Popular del Cesar

SVM: Máquinas vectoriales de soporte

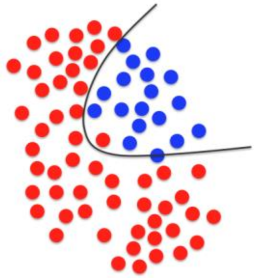
1) Lineal con separación perfecta



2) Lineal sin separación perfecta

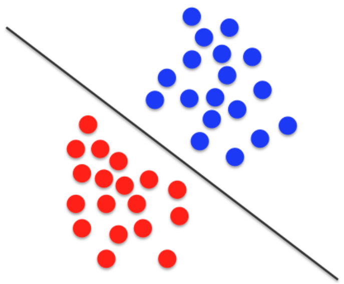


3) No lineal



SVM: Máquinas vectoriales de soporte

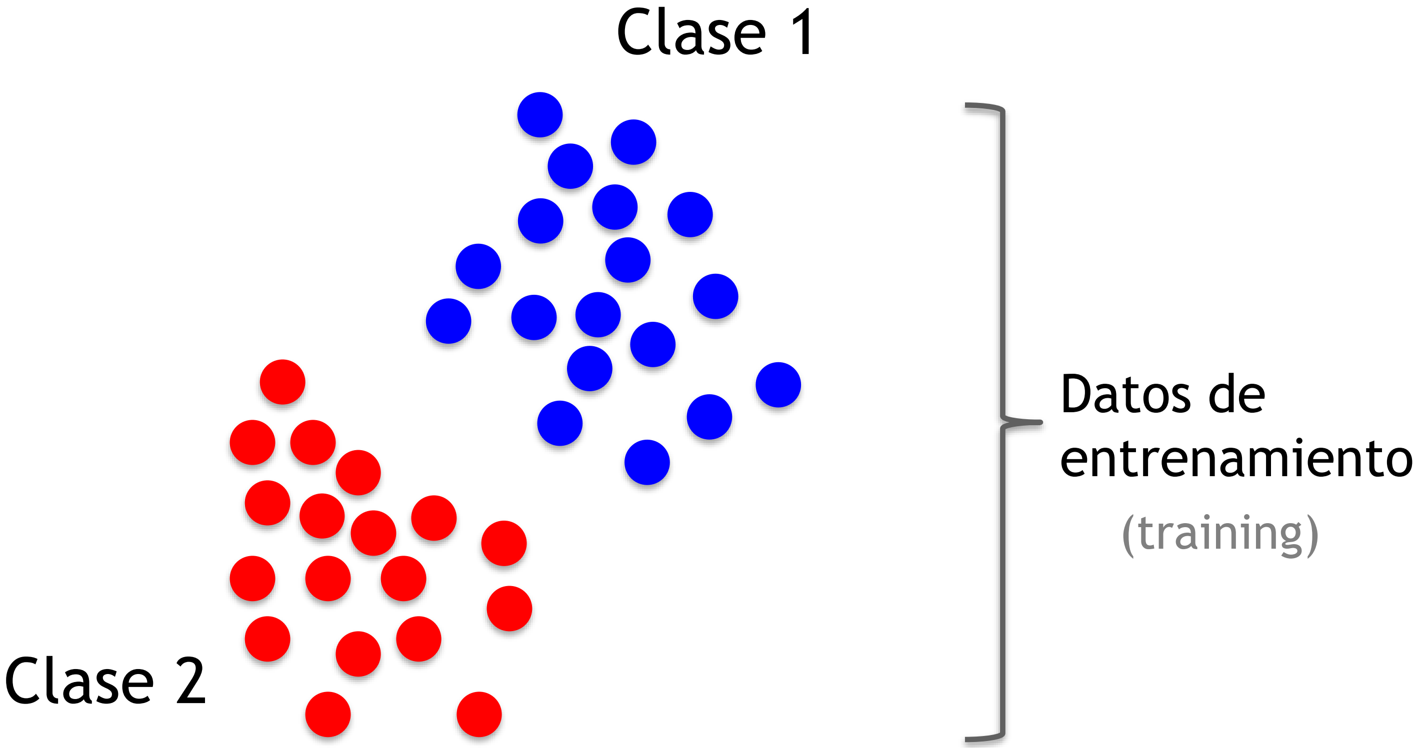
1) Lineal con separación perfecta



2) Lineal sin separación perfecta

3) No lineal

SVM: dos clases



SVM: dos clases

La línea 1 separa perfectamente las clases

La Línea 2 separa las clases perfectamente

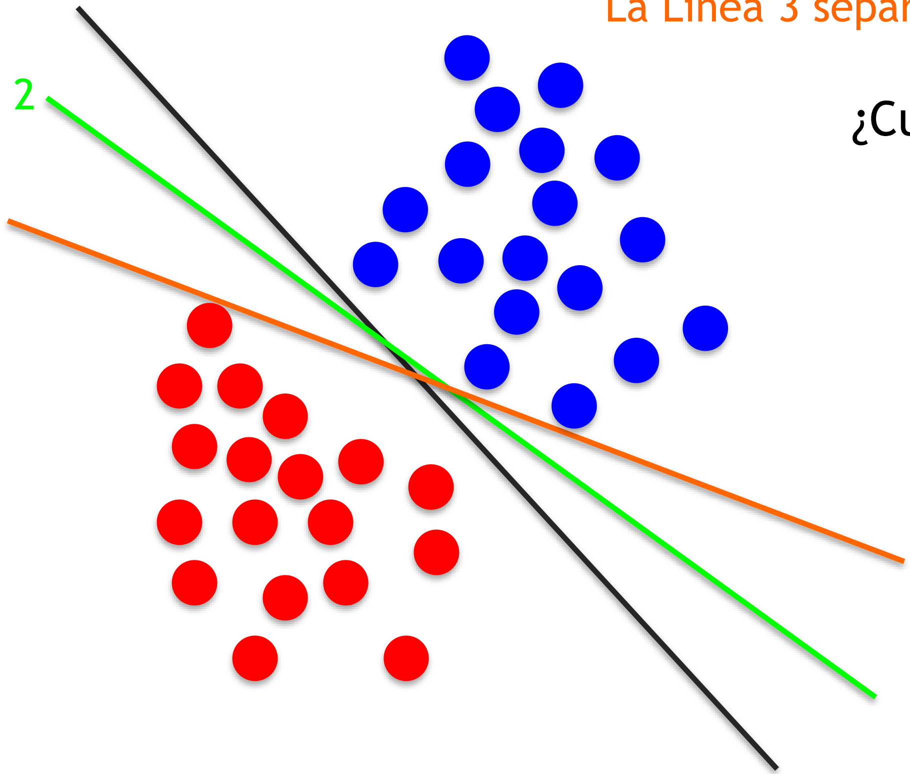
La Línea 3 separa las clases perfectamente

¿Cuál es mejor?

Línea de decisión 1

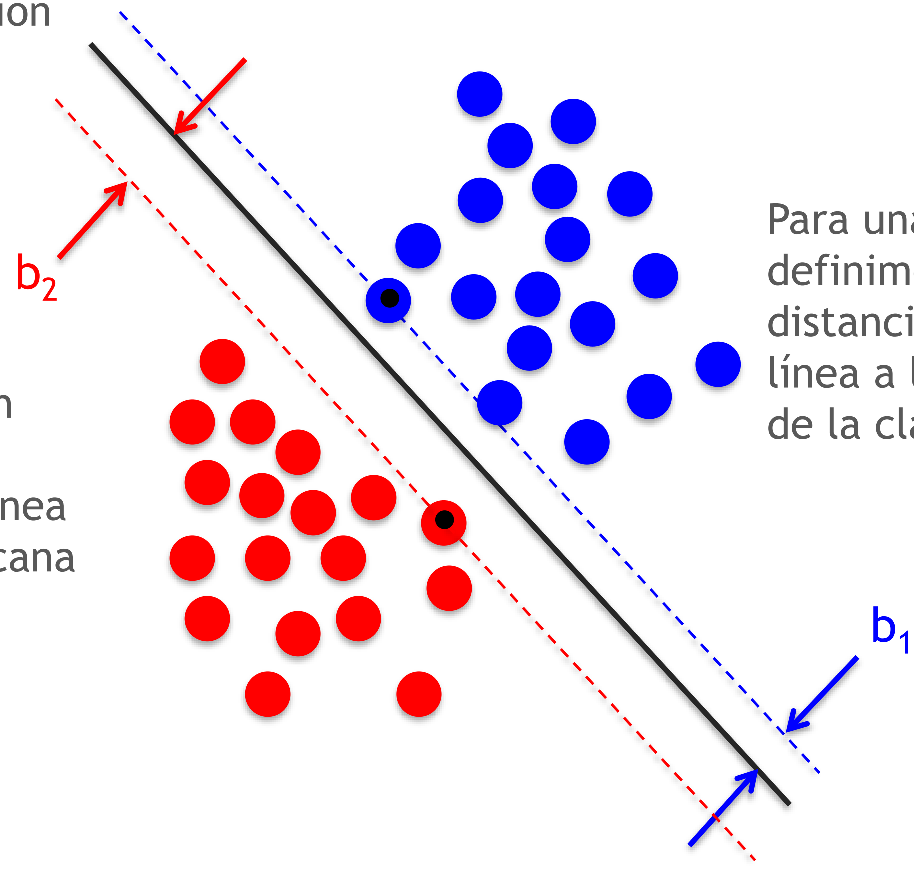
Línea de decisión 2

Línea de decisión 3



SVM: dos clases

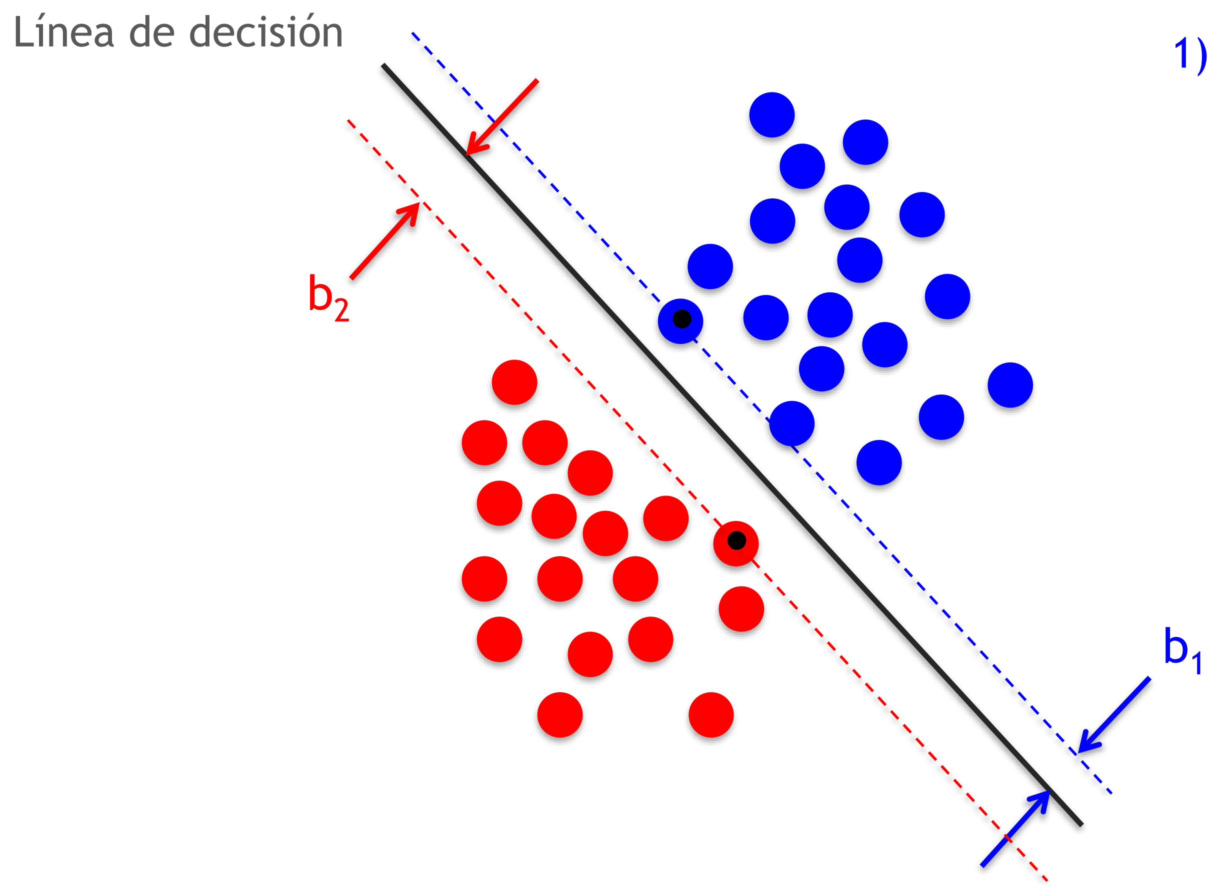
Línea de decisión



Para una línea de decisión, definimos el margen 'b1' como la distancia perpendicular de la línea a la muestra más cercana de la clase 1.

Y definimos el margen 'b2' como la distancia perpendicular de la línea a la muestra más cercana de la clase 2.

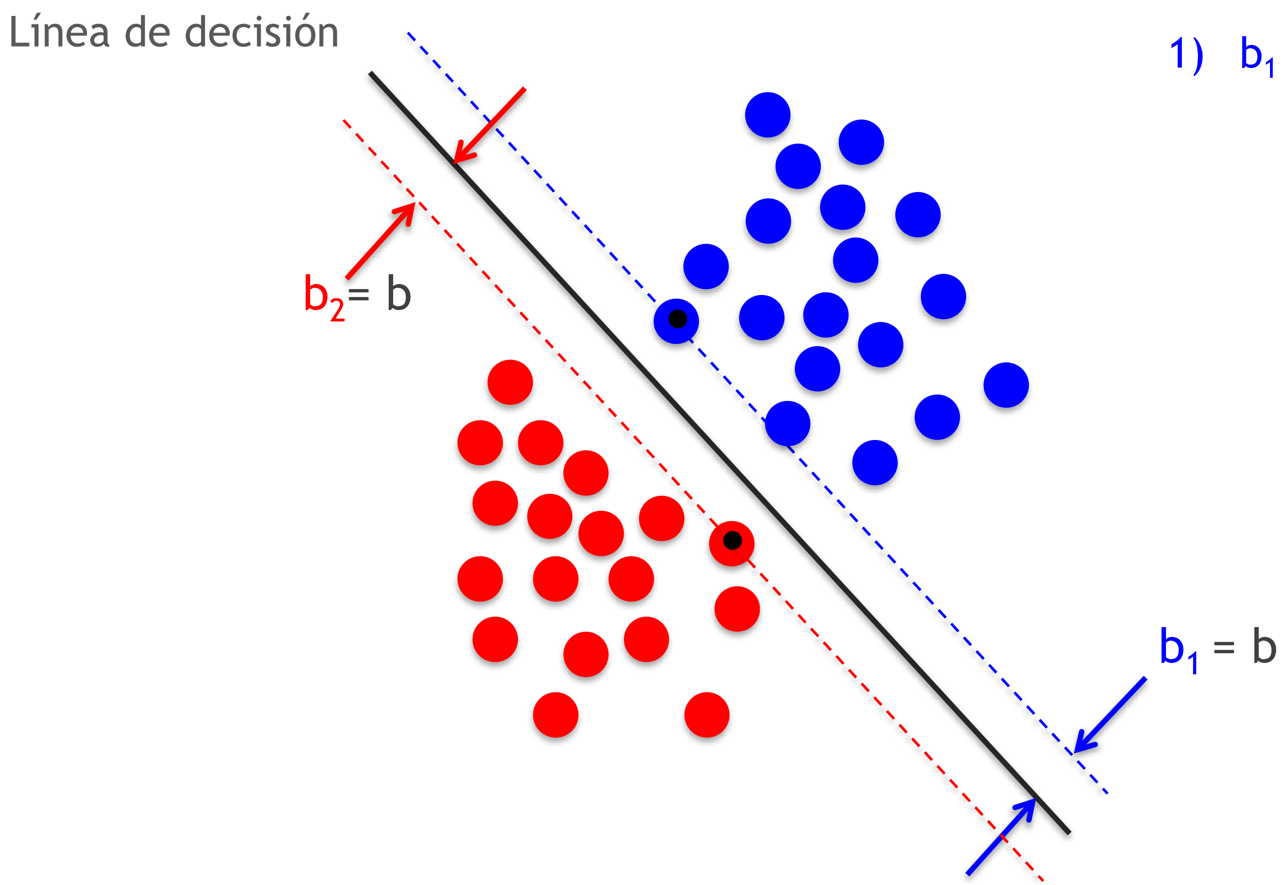
SVM: dos clases



Ideas clave de SVM:

1) $b_1 = b_2 = b$.

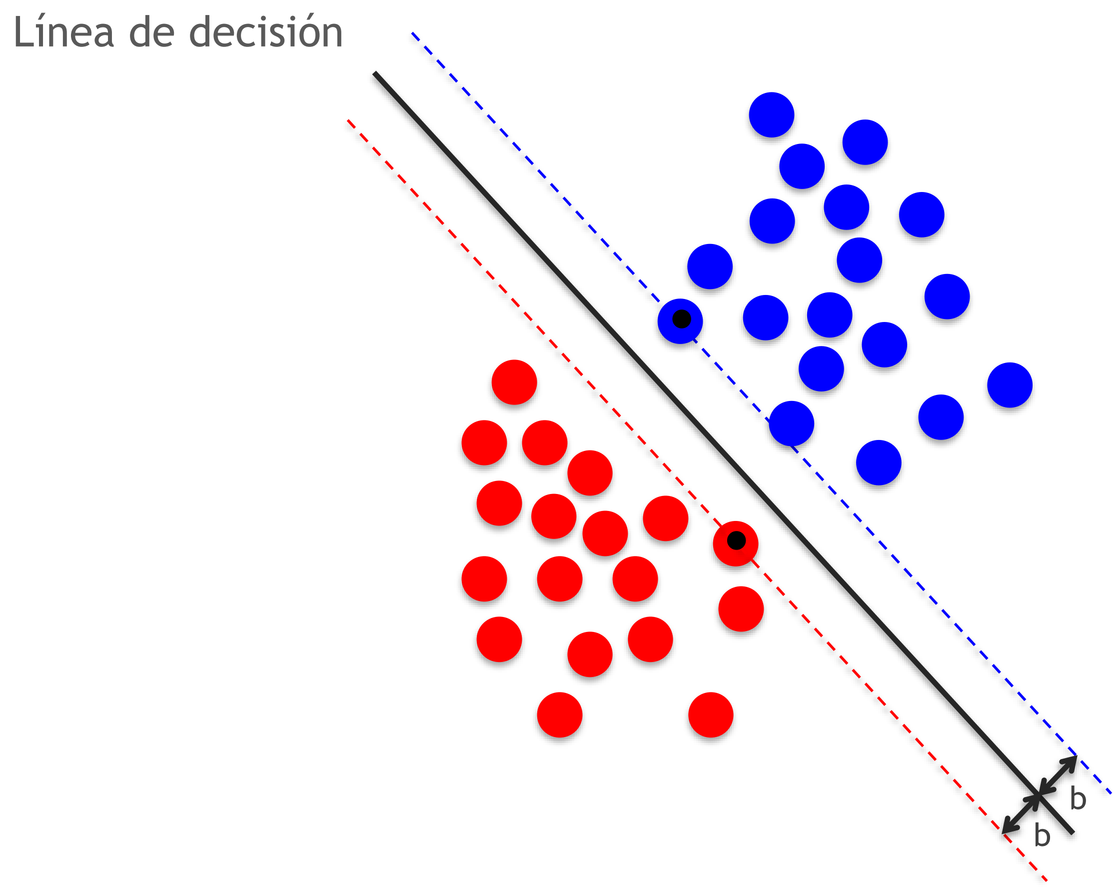
SVM: dos clases



Ideas clave de SVM:

1) $b_1 = b_2 = b$.

SVM: dos clases

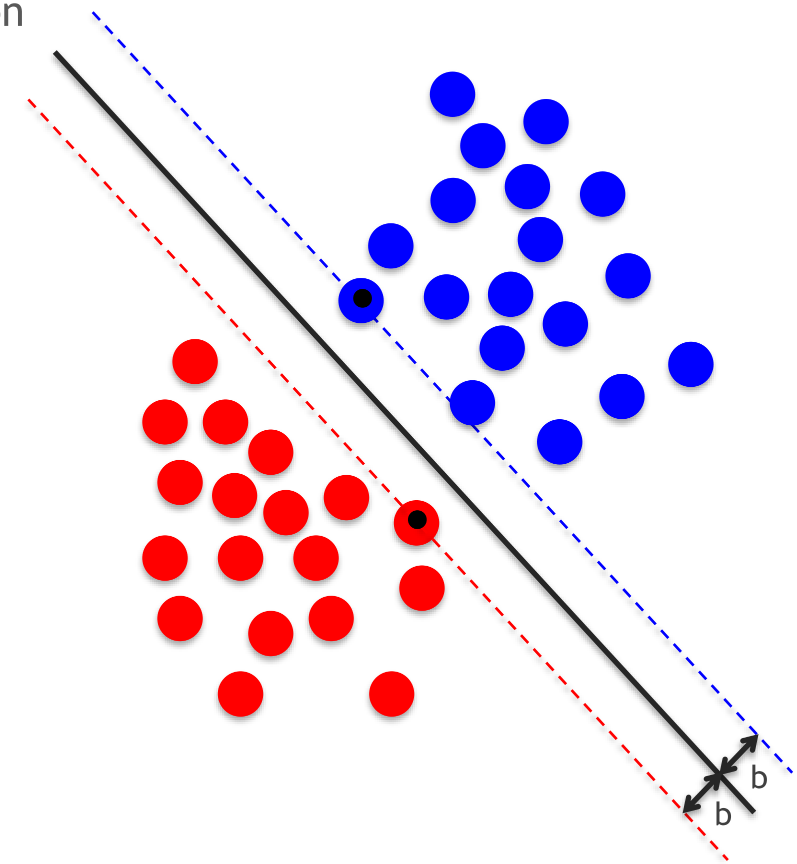


Ideas clave de SVM:

1) $b_1 = b_2 = b$.

SVM: dos clases

Línea de decisión

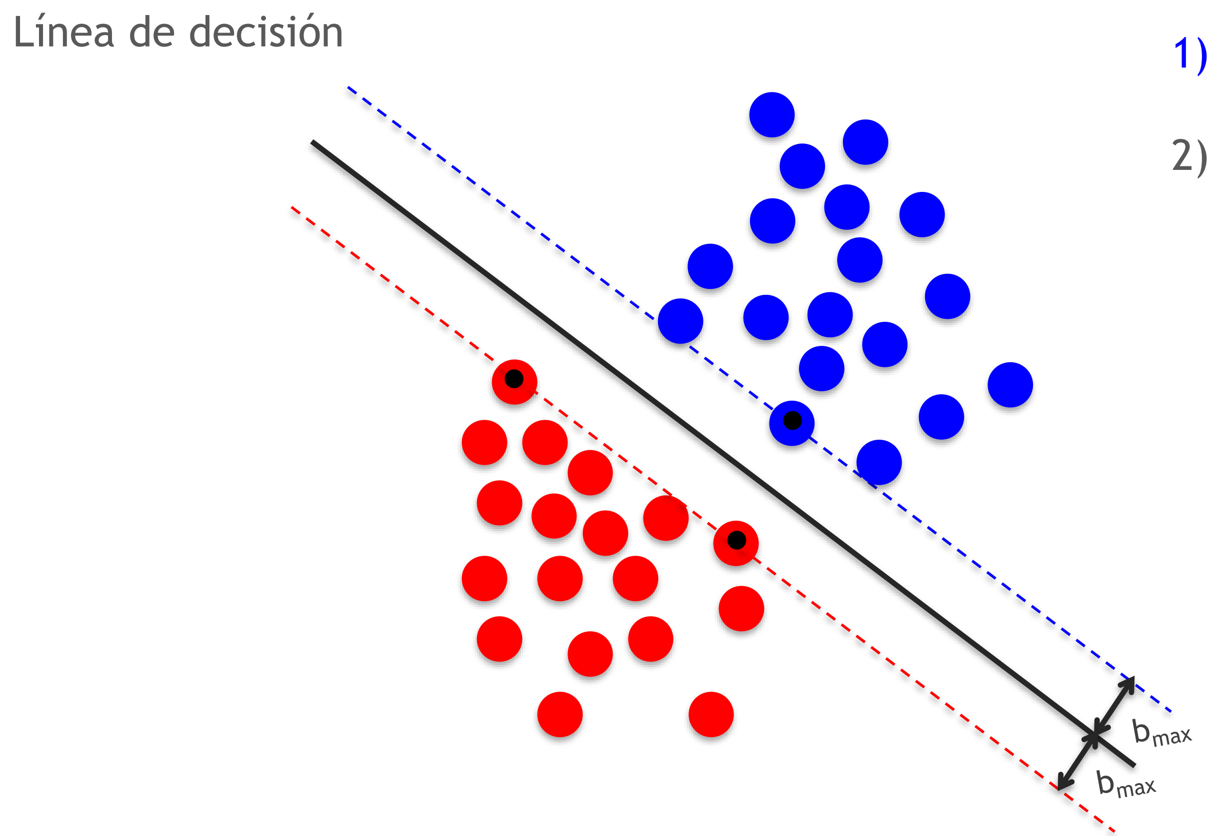


Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.

SVM: dos clases

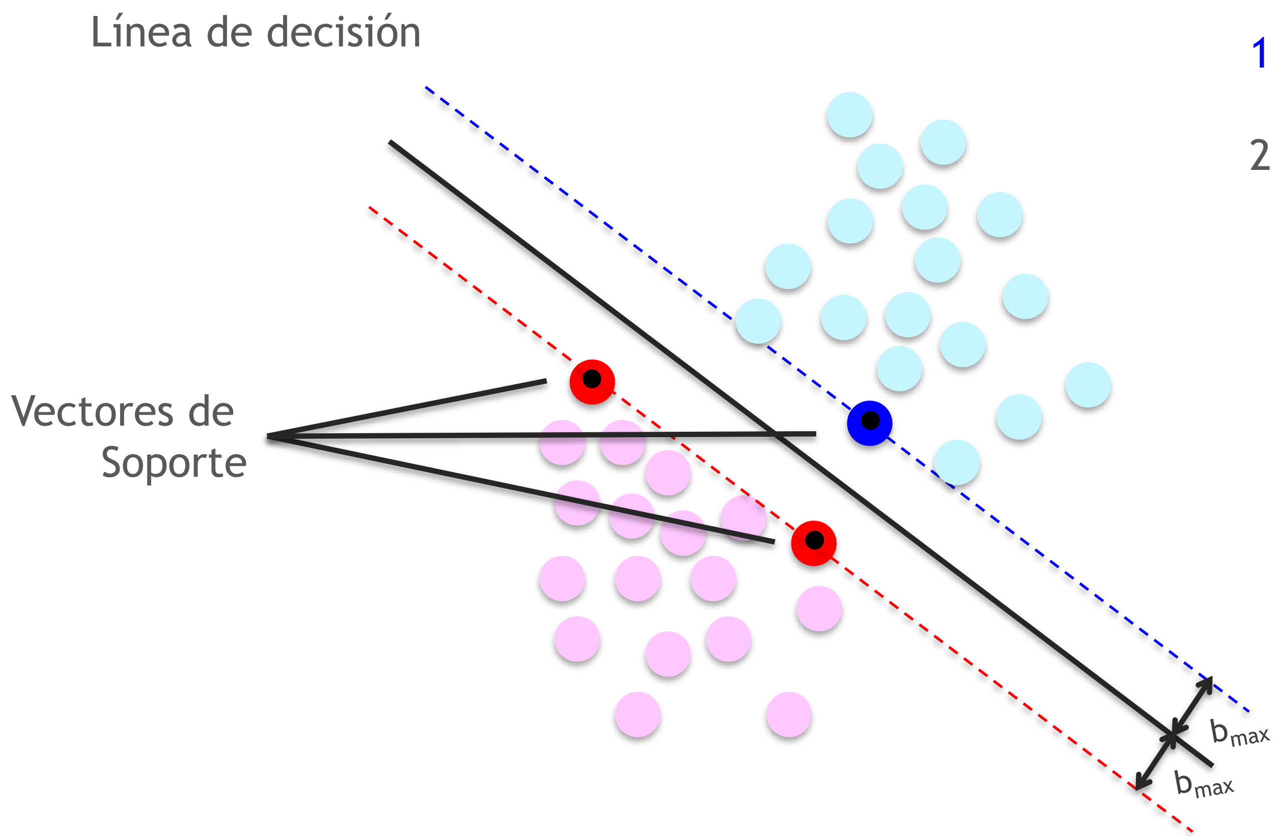


Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.

SVM: dos clases

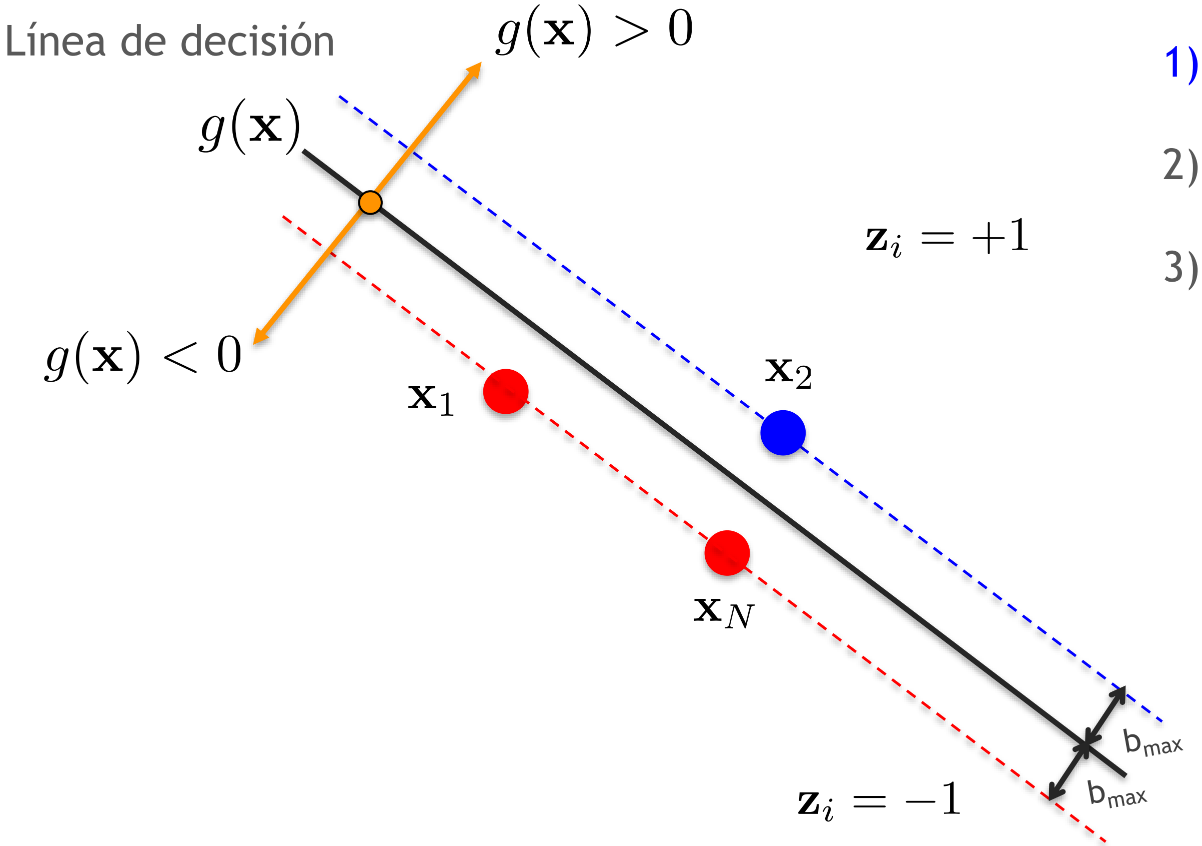


Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.

SVM: dos clases



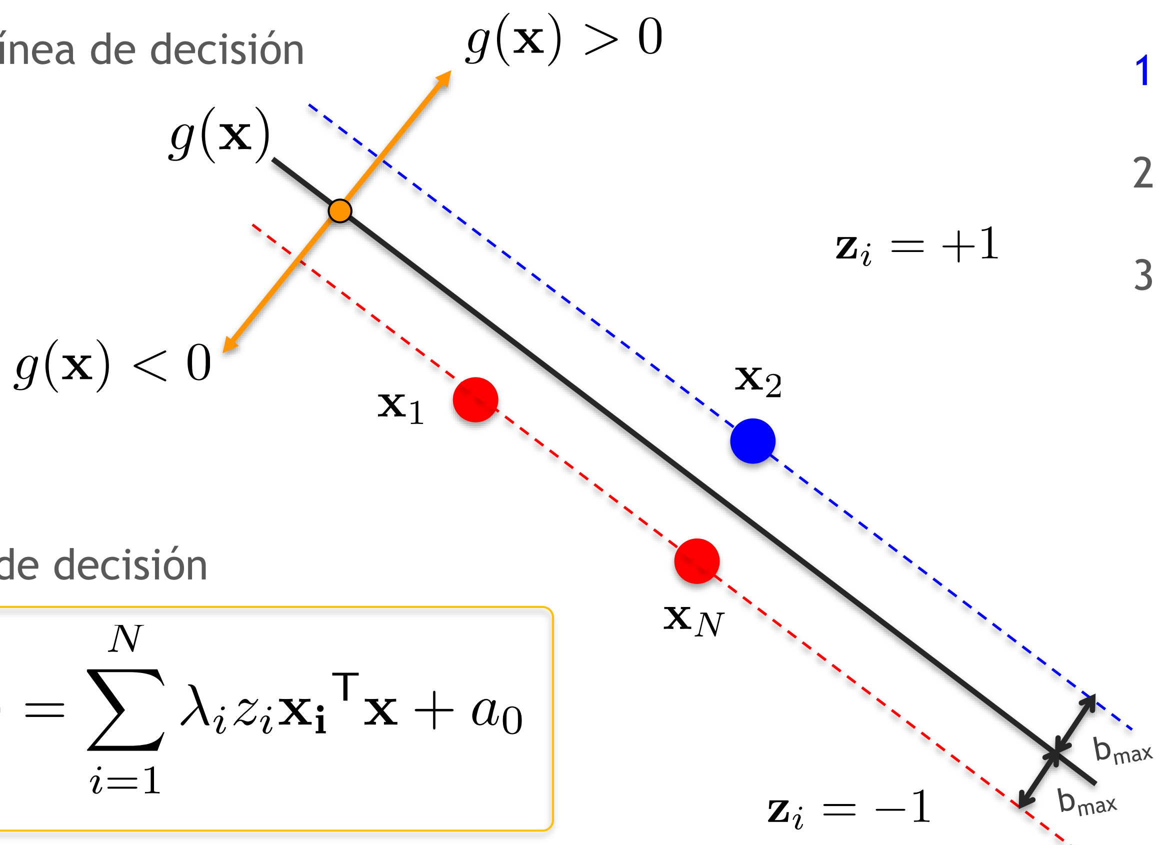
Ideas clave de SVM:

1) $b_1 = b_2 = b$.

2) b debe maximizarse.

3) Solución: $g(\mathbf{x})$

SVM: dos clases



Ideas clave de SVM:

1) $b_1 = b_2 = b$.
2) b debe maximizarse.
3) Solución: $g(\mathbf{x})$

Línea de decisión

$$g(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i z_i \mathbf{x_i}^{\mathsf{T}} \mathbf{x} + a_0$$

La solución de SVM

TRAINING:

Datos de Entrenamiento
$(\mathbf{x}_k, \mathbf{z}_k)$

características

+1 or -1
(2 clases)

Optimización

Número de vectores de soporte

constantes

constante

línea de decisión

Muestra de prueba

$$g(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i \underbrace{z_i \mathbf{x_i}}_{\text{Datos de entrenamiento para los vectores de soporte}}{}^{\mathsf{T}} \mathbf{x} + a_0$$

TESTING: si $g(\mathbf{x}) > 0$ entonces clase = +1
en caso contrario clase = -1

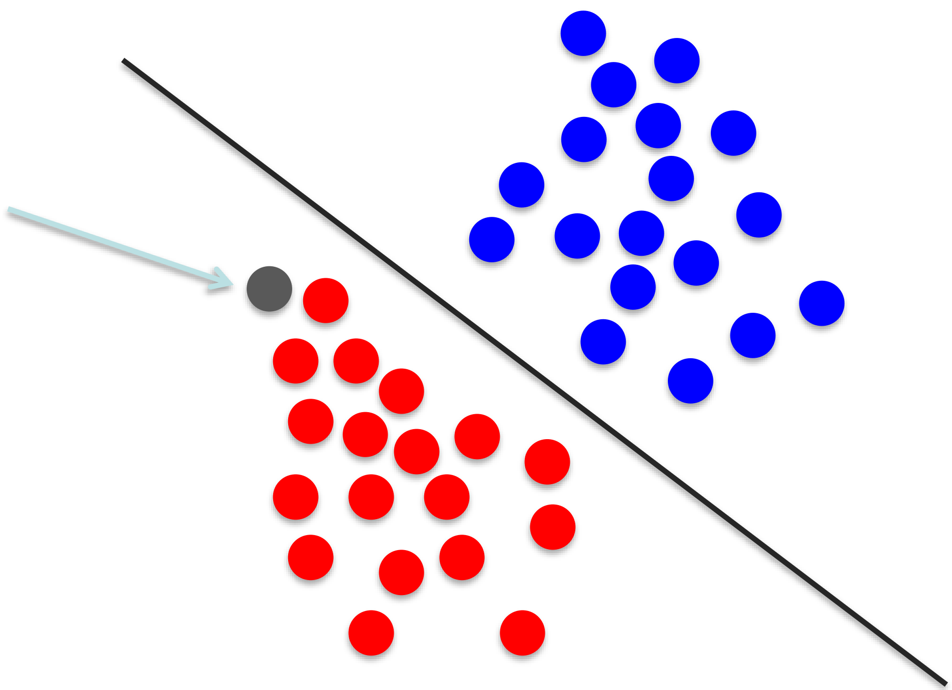
SVM: dos clases

Línea de decisión

(testing)

Dato de prueba

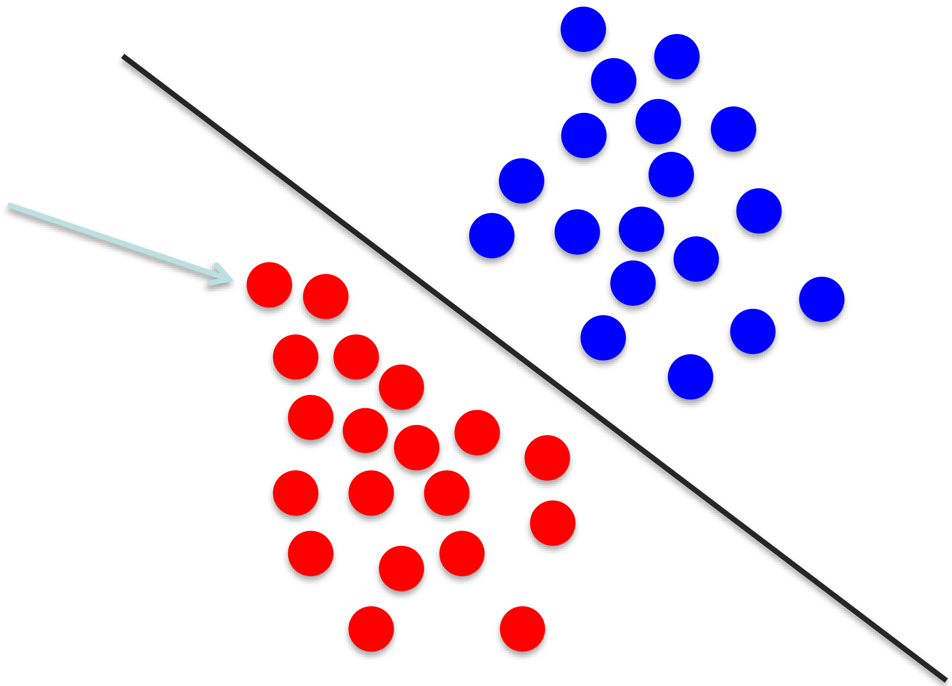
?



SVM: dos clases

Línea de decisión

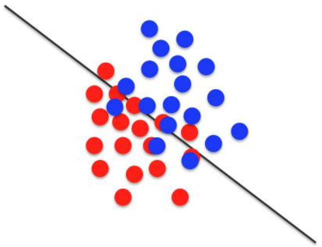
(testing)
Dato de prueba
?



SVM: Máquinas vectoriales de soporte

1) Lineal con separación perfecta

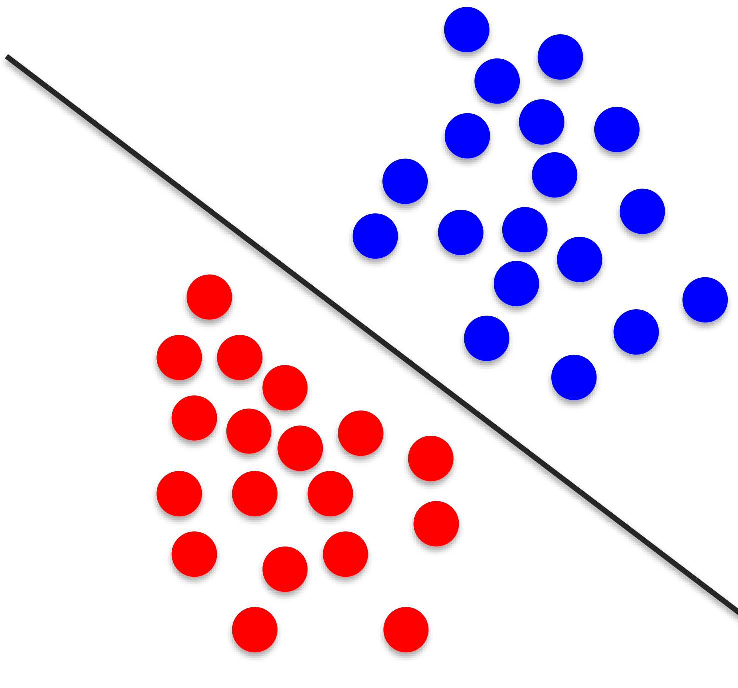
2) Lineal sin separación perfecta



3) No lineal

SVM: dos clases

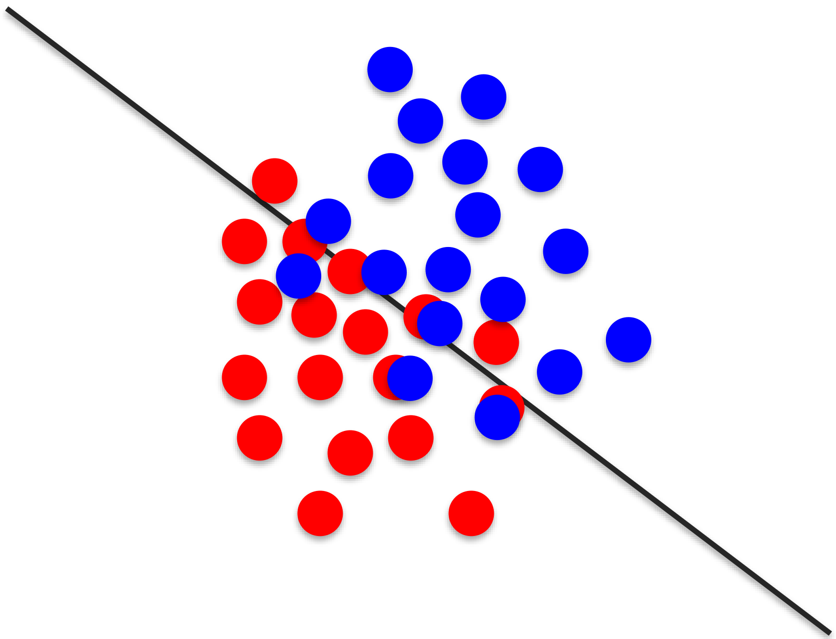
¿Cómo definir la línea de decisión cuando no hay una separación perfecta?



SVM: dos clases

¿Cómo definir la línea de decisión cuando no hay una separación perfecta?

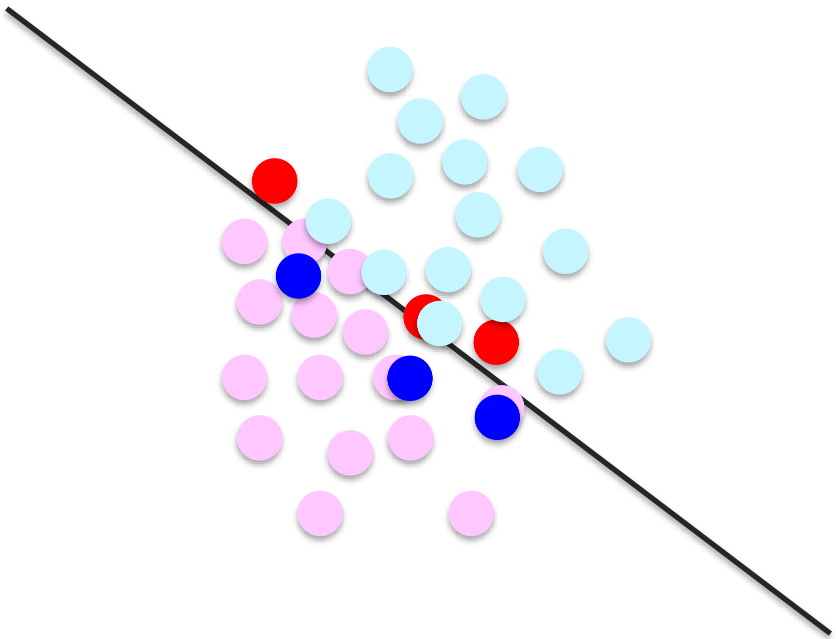
Consideramos sólo las muestras clasificadas erroneamente



SVM: dos clases

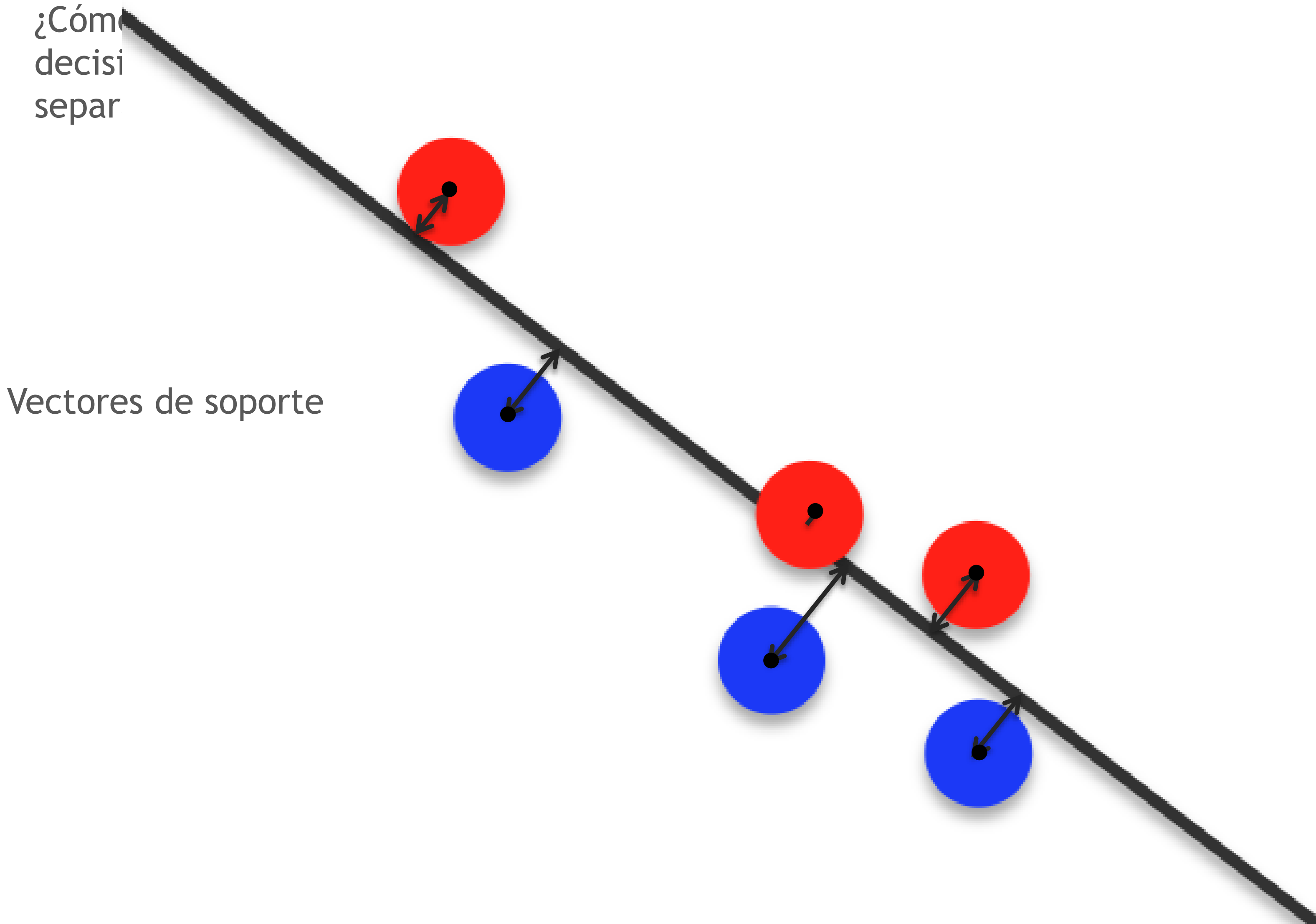
¿Cómo definir la línea de decisión cuando no hay una separación perfecta?

Consideramos sólo las muestras clasificadas erroneamente



SVM

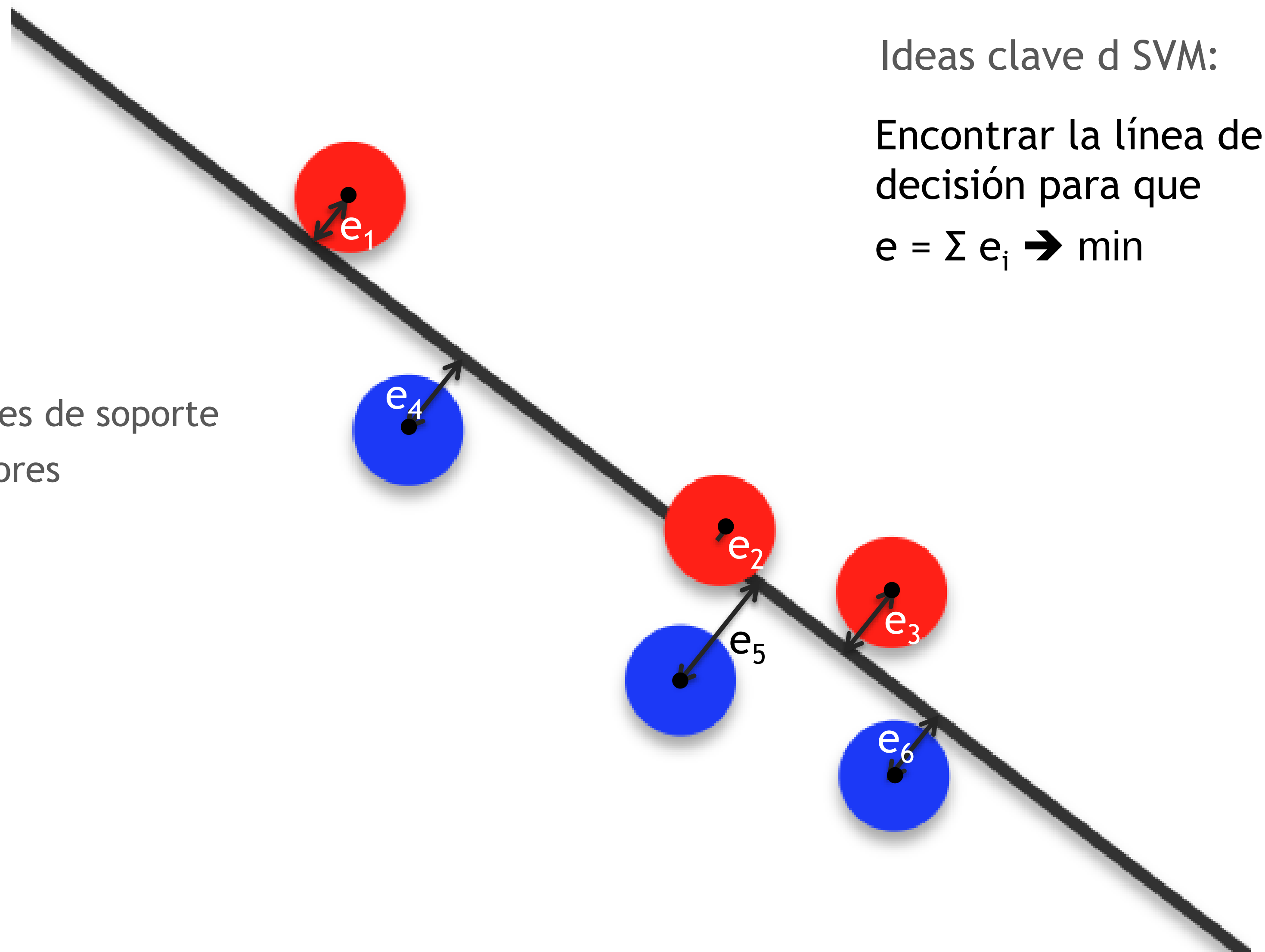
¿Cóm muestras
decisi ente
separ



Ideas clave d SVM:

Encontrar la línea de decisión para que

$e = \Sigma e_i \rightarrow \min$



Vectores de soporte

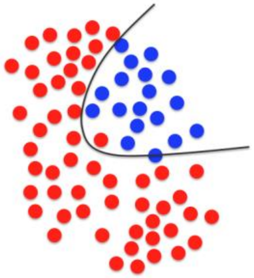
e_i: errores

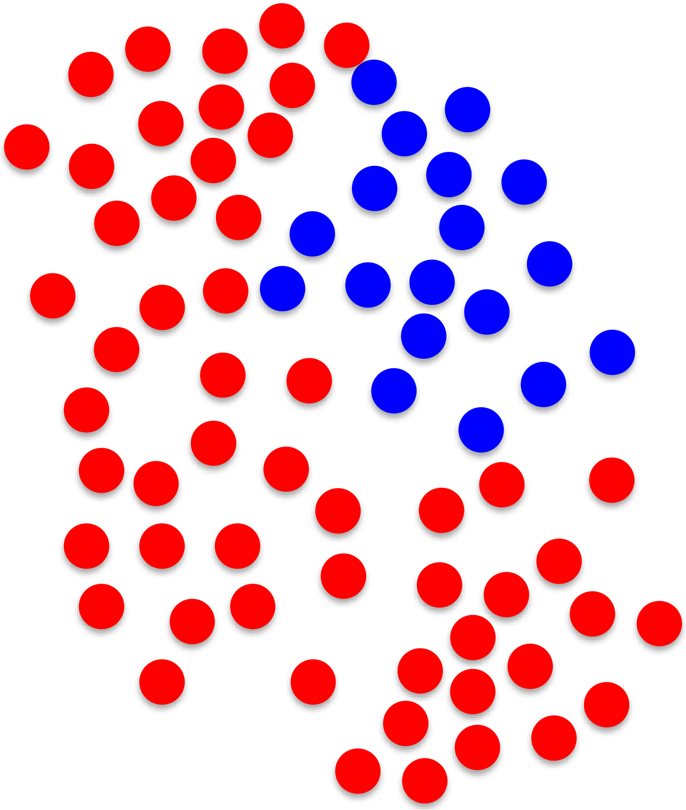
SVM: Máquinas vectoriales de soporte

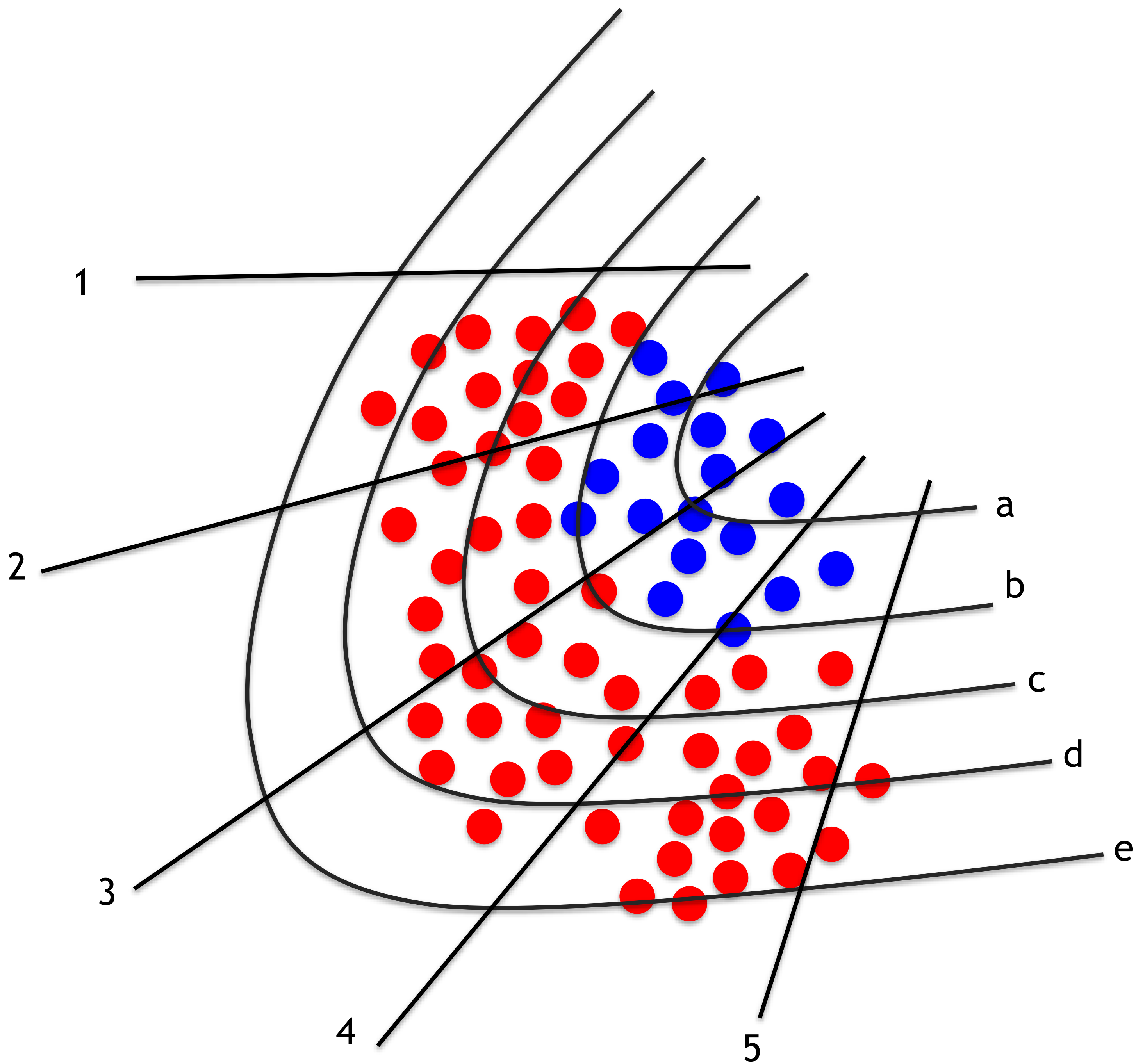
1) Lineal con separación perfecta

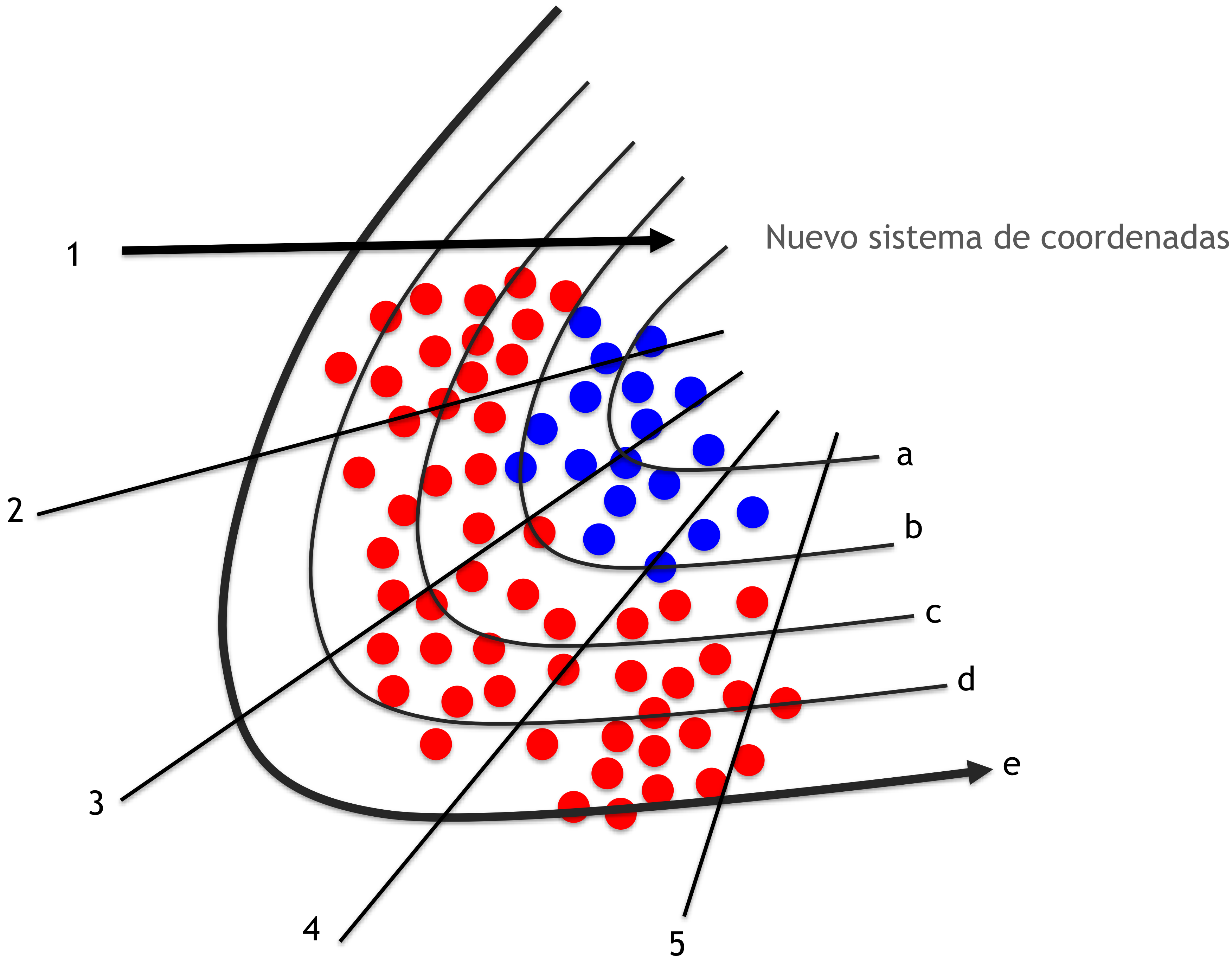
2) Lineal sin separación perfecta

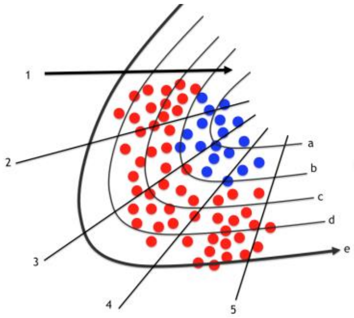
3) No lineal



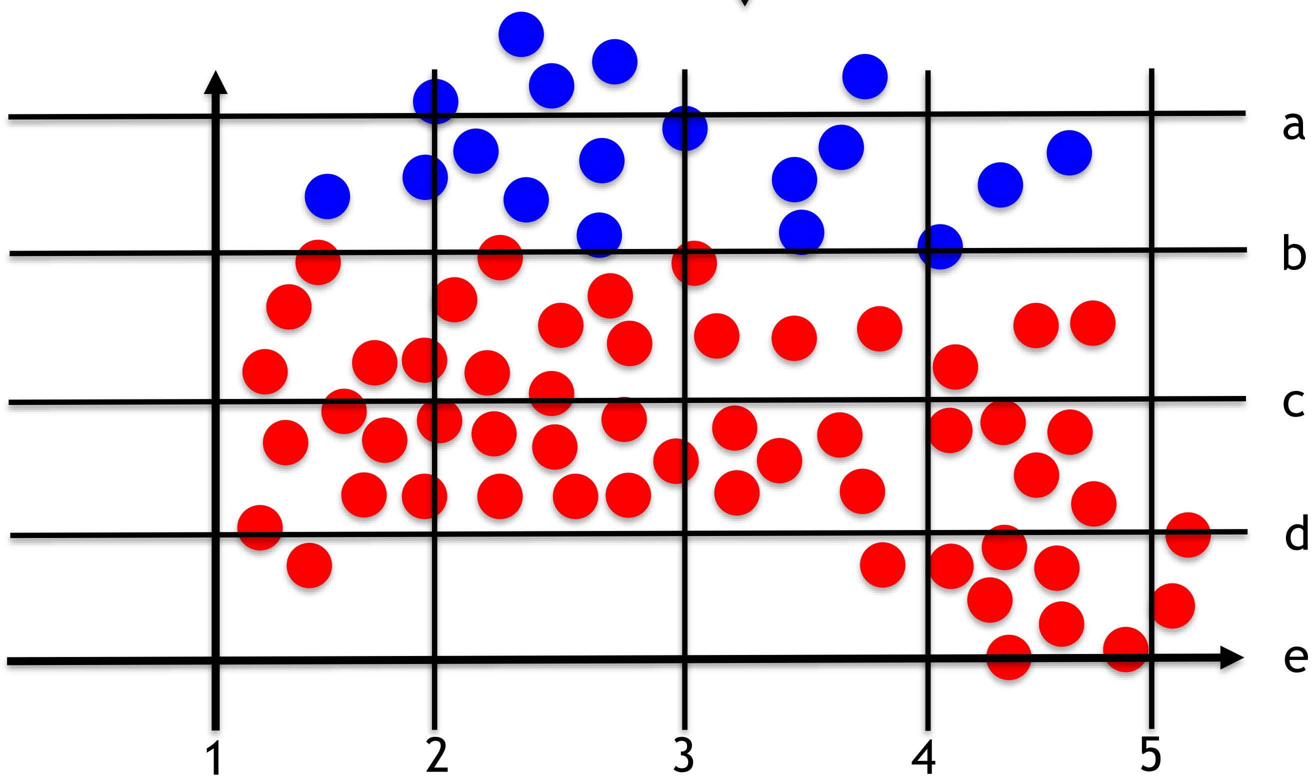


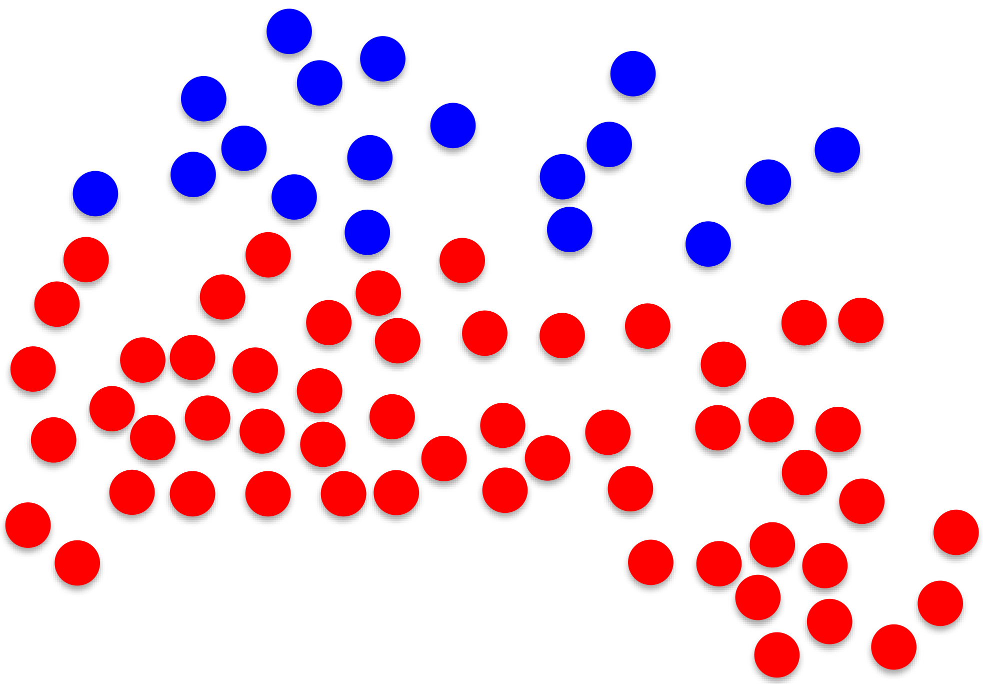

1
2
3
4
5
a
b
c
d
e


Nuevo sistema de coordenadas
1
2
3
4
5
a
b
c
d
e

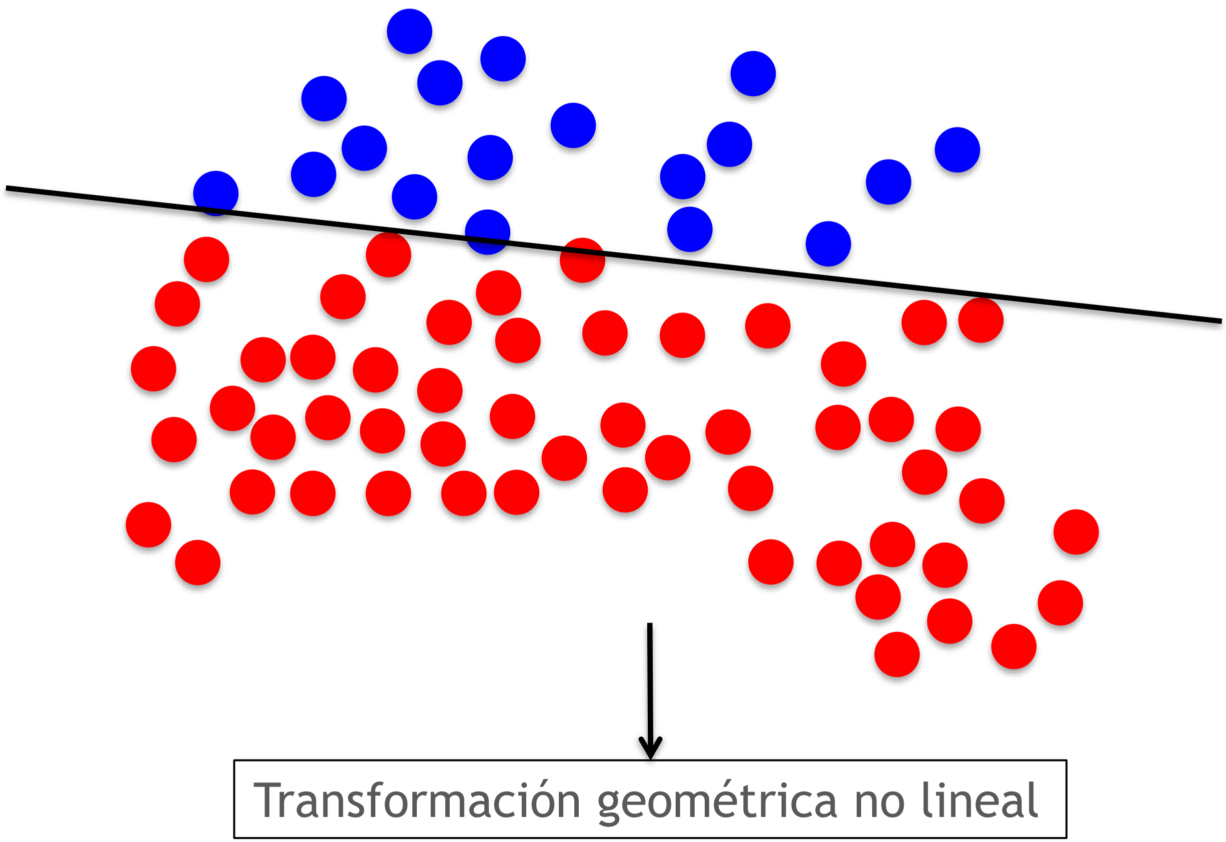


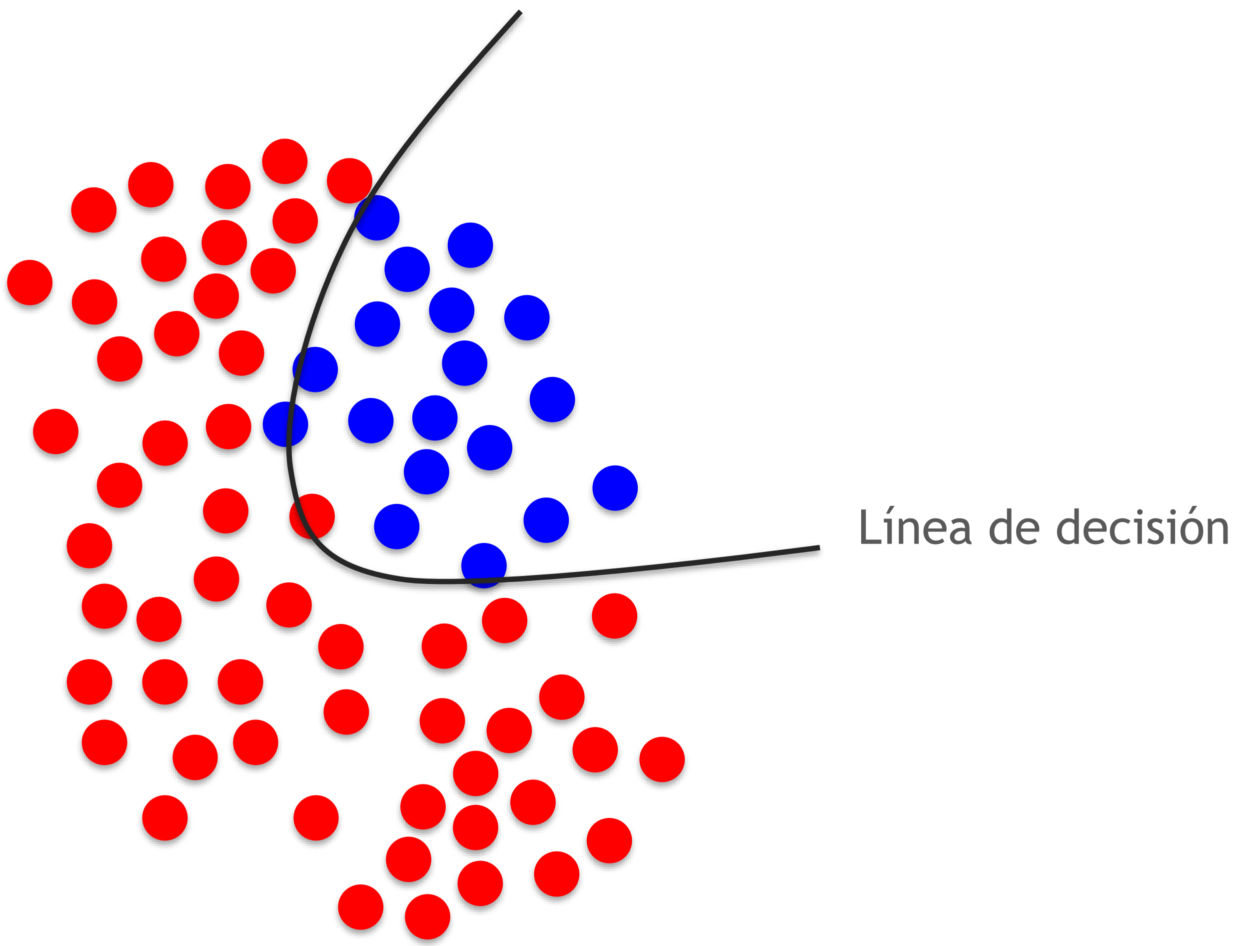
Transformación geométrica no lineal





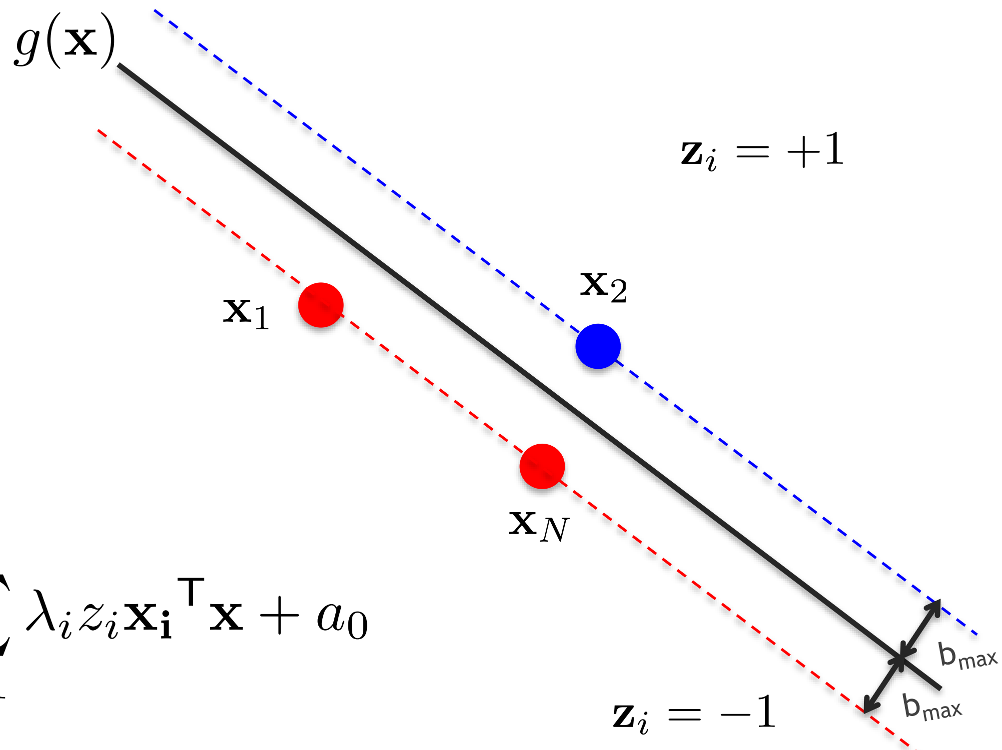
SVM lineal en nuevo sistema de coordenadas




Línea de decisión

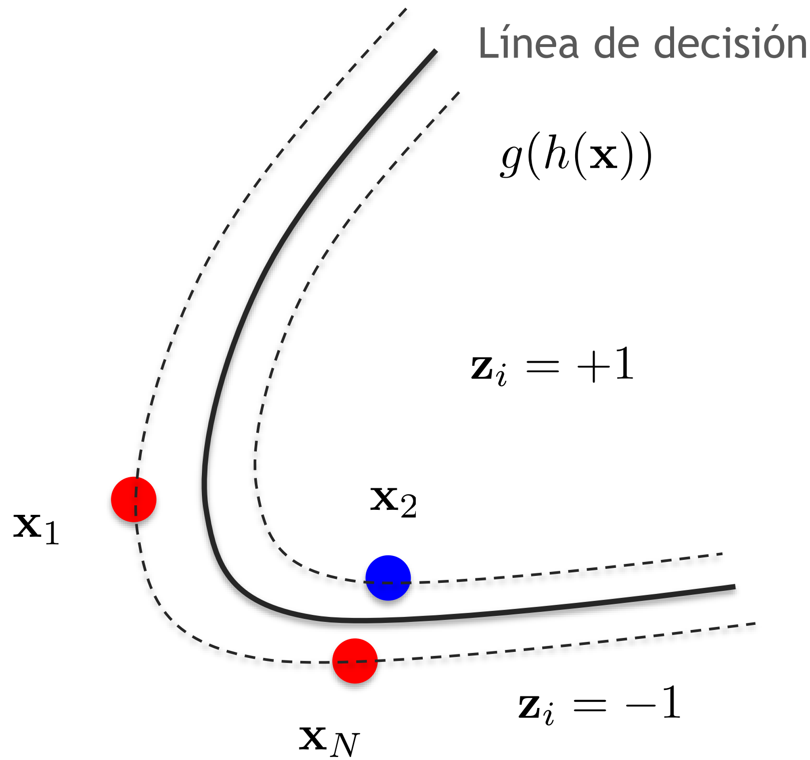
SVM: Máquinas vectoriales de soporte

Línea de decisión



$$g(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i z_i \mathbf{x_i}^{\mathsf{T}} \mathbf{x} + a_0$$

SVM: Máquinas vectoriales de soporte



$$g(h(\mathbf{x})) = \sum_{i=1}^{N} \lambda_i z_i h(\mathbf{x_i})^{\mathsf{T}} h(\mathbf{x}) + a_0$$

$$g(h(\mathbf{x})) = \sum_{i=1}^{N} \lambda_i z_i < h(\mathbf{x_i}), h(\mathbf{x}) > + a_0$$

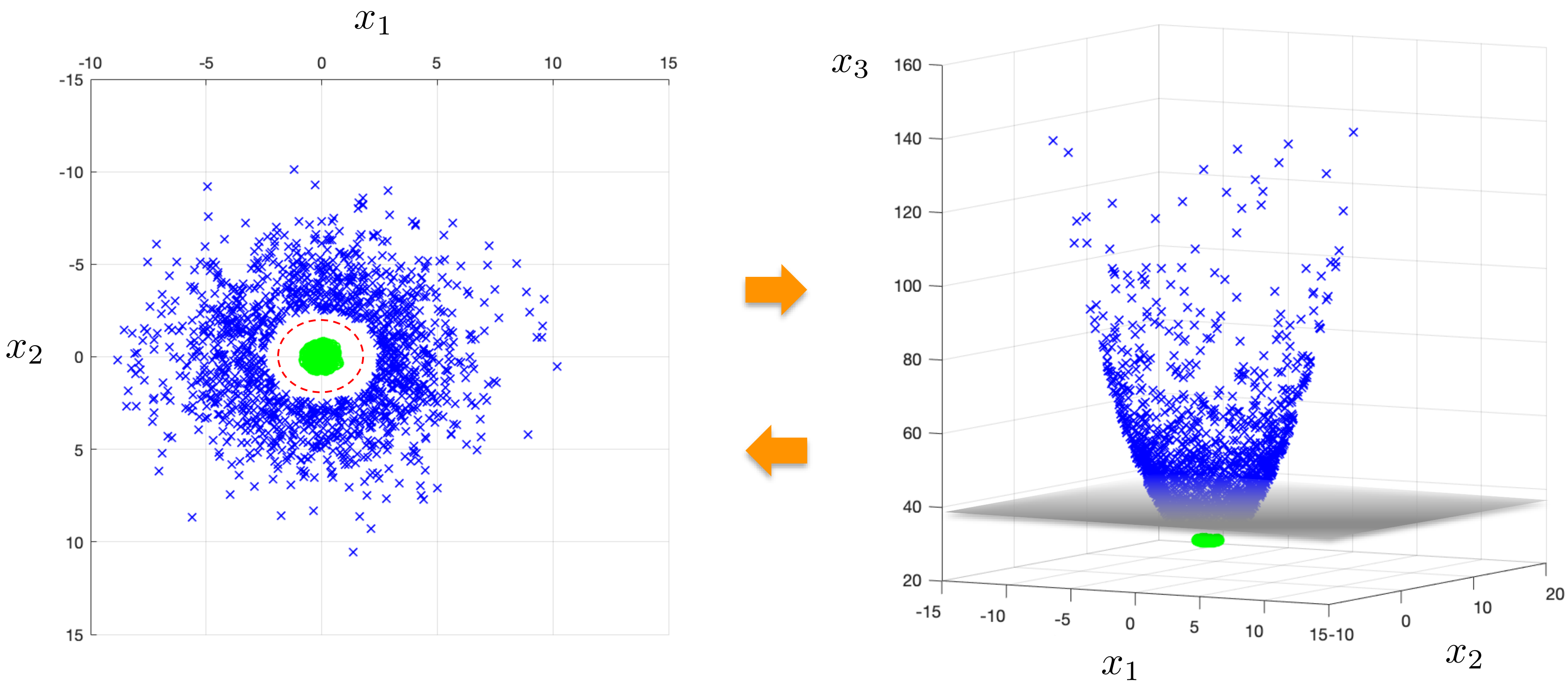
No se necesita $h(\mathbf{x})$, solo es necesario el kernel $< h(\mathbf{x_i}), h(\mathbf{x}) >$

SVM: Kernels

$$K(\mathbf{x}', \mathbf{x}) = \langle h(\mathbf{x}'), h(\mathbf{x}) \rangle =$$

linear	$\langle \mathbf{x}', \mathbf{x} \rangle$
polynomial	$(1 + \langle \mathbf{x}', \mathbf{x} \rangle)^n$
radial basis	$\exp(-\|\|\mathbf{x}' - \mathbf{x}\|\|^2 / c)$
sigmoid	$\tanh(K_1 \langle \mathbf{x}', \mathbf{x} \rangle + K_2)$

SVM: El truco del Kernel



La separación lineal es imposible

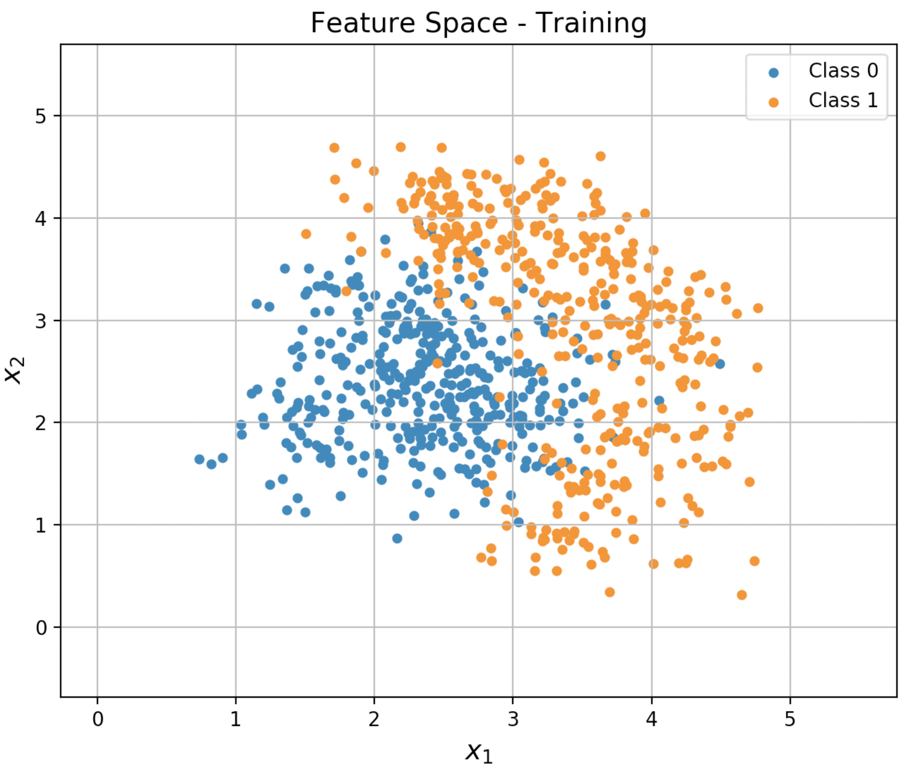
La separación lineal es perfecta

Ejemplos

Ejemplo

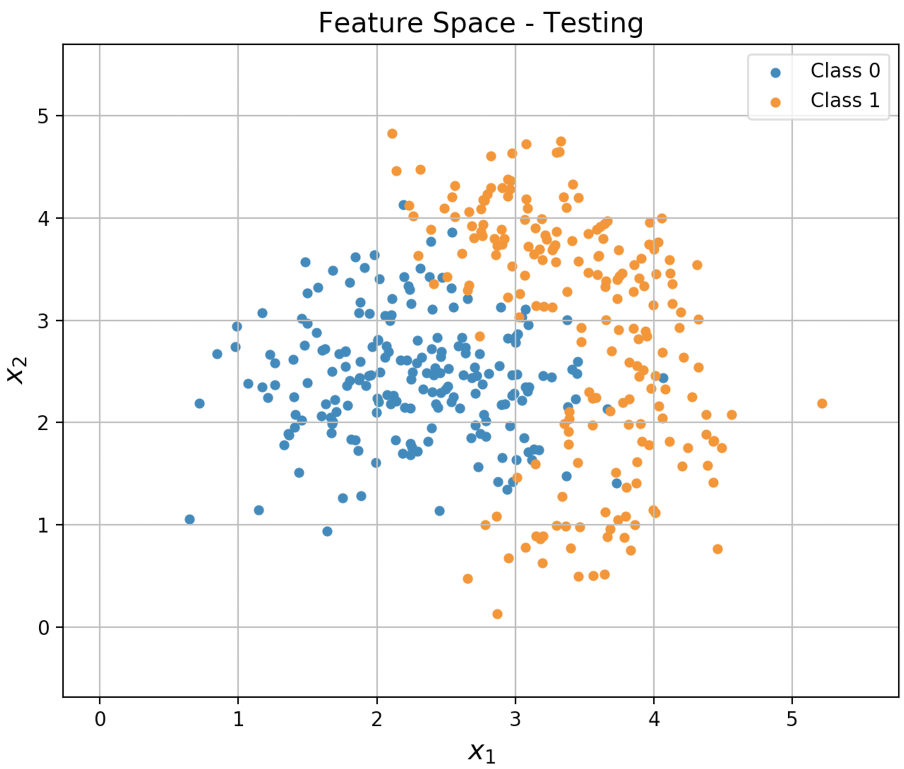
(X,d)

TRAINING



(X_t,d_t)

TESTING

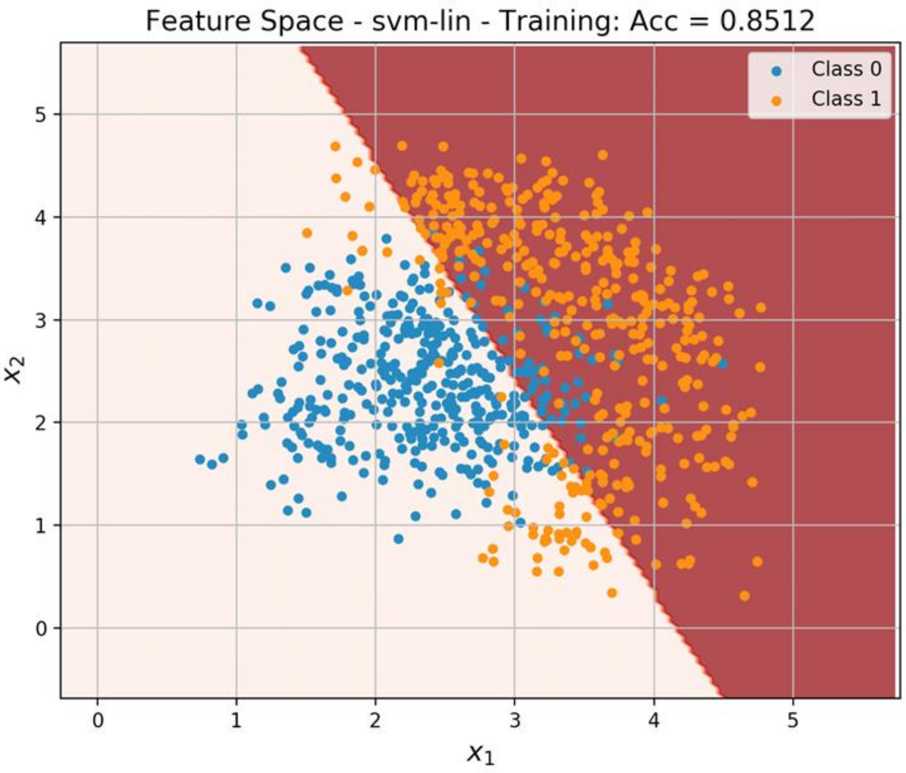


Ejemplo

SVM-LIN

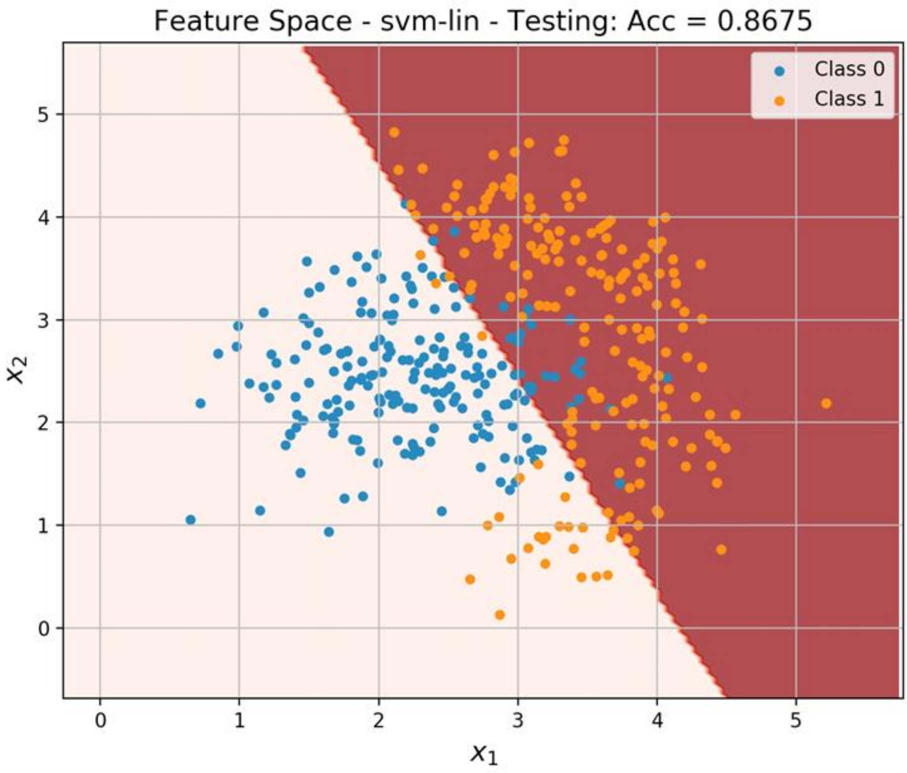
(X,d)

TRAINING



(X_t,d_t)

TESTING

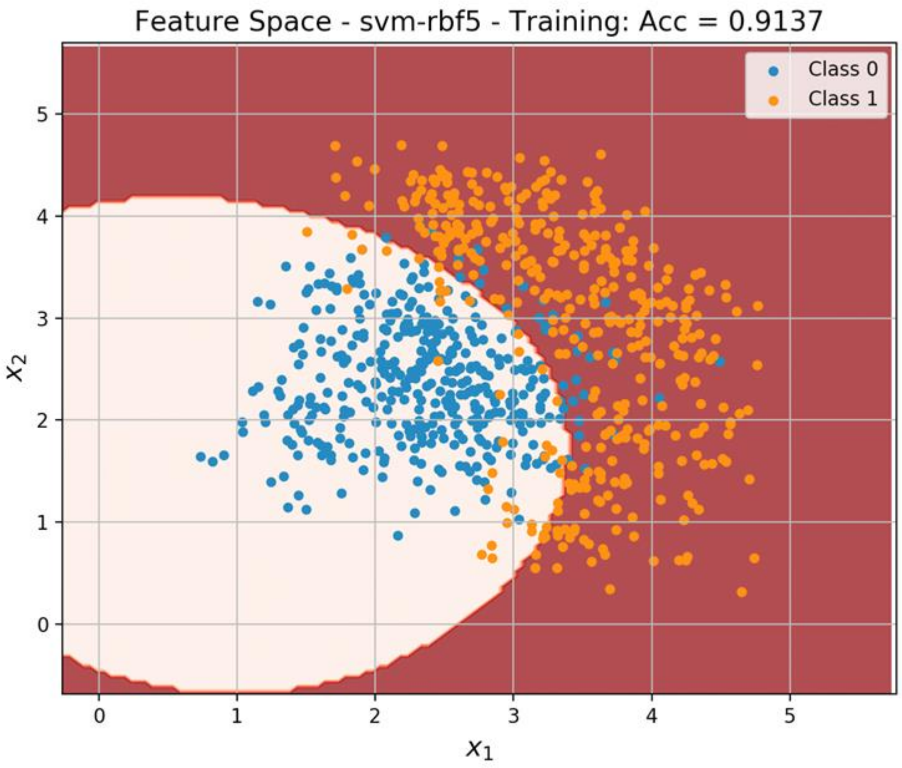


Ejemplo

SVM-RBF

(X,d)

TRAINING



(X_t,d_t)

TESTING